

Vysoká škola báňská – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Absolvování individualní odborné praxe
Individual Professional Practice in the Company

2015

Nikola Vrbinářová

Zadání bakalářské práce

Student:

Nikola Vrbinárová

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R059 Mobilní technologie

Téma:

Absolvování individuální odborné praxe
Individual Professional Practice in the Company

Zásady pro vypracování:

1. Student vykoná individuální praxi ve firmě: ELCOM, a.s.
2. Struktura závěrečné zprávy:
 - a. Popis odborného zaměření firmy, u které student vykonal odbornou praxi a popis pracovního zařazení studenta
 - b. Seznam úkolů zadaných studentovi v průběhu odborné praxe s vyjádřením jejich časové náročnosti
 - c. Zvolený postup řešení zadaných úkolů
 - d. Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe
 - e. Znalosti či dovednosti scházející studentovi v průběhu odborné praxe
 - f. Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení

Seznam doporučené odborné literatury:

Podle pokynů konzultanta, který vedl odbornou praxi studenta

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Zdeňka Chmelíková, Ph.D.**

Konzultant bakalářské práce: Jiří Hula, Ing., Ph.D.

Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015



doc. Ing. Miroslav Vozňák, Ph.D.
vedoucí katedry




prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení studenta

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne: *4. mája 2015*


.....
podpis studenta

Prohlášení zástupce spolupracující právnické nebo fyzické osoby

„Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava.“

Dne: 23. *apríla* 2015

.....
podpis zástupce



Poděkování

Rada by som sa poďakovala môjmu konzultantovi Ing. Jiřímu Hulovi, Ph.D. a kolegom vo firme ELCOM, a.s., za odbornú pomoc a konzultácie pri vytváraní tejto bakalárskej práce. Taktiež moja vďaka patrí pani Ing. Zdenke Chmelíkové, Ph.D. za vedenie mojej práce a poskytnuté rady a pripomienky.

Abstrakt

Bakalárska práca je zameraná na pracovnú činnosť a riešenie zadaných úloh počas odbornej praxe vykonávanej vo firme ELCOM, a.s.. Obsah tvorí i popis jednotlivých prostriedkov a nástrojov, ktoré som využívala pri spracovaní zadaných úloh a dosiahnuté výsledky spolu s hodnotením prínosu praxe.

Cieľom praxe bolo získať potrebné znalosti z vývojového prostredia LabVIEW a uplatniť ich pri riešení zadaných úloh. Po zvládnutí základov daného programovacieho jazyka pristúpiť k riešeniu projektu tvorby ovládača pre servopohon. Práca na projekte zahŕňala oboznámenie sa s tvorbou ovládačov, prácu s komunikačnými protokolmi a štúdium príslušnej dokumentácie. Nasledovala postupná tvorba ovládača, hľadanie správnych postupov pri tvorbe a realizácii, tvorba príkladov pre prácu s ovládačom a vypracovanie dokumentácie. Poslednou zadanou úlohou bola štúdia realizovateľnosti, zameraná na určenie vhodnej metódy pre detekovanie chýb v svetlovodoch.

Kľúčová slova

ELCOM, a.s.; National Instruments; LabVIEW; servopohon; prístrojový ovládač; Modbus; TCP/IP; WorkBench; svetlovod

Abstract

This bachelor thesis is focused on my work activity and working issues in realized projects during my professional work practice in firm ELCOM, a.s. It contains also the description of the tools and environment, which were used to create solutions for my working tasks and my achieved results during working practice.

The aim of my practice was to acquire skills in programming environment LabVIEW and be able to use my knowledge in order to solve my working tasks. After handling the basics of language I was able to start working on the project of driver for servo drive. The work on the project required study of creating drivers, communication protocols and documentation. The next step was creating the driver for servo drive, looking for solutions in actual issues and realization, creation of examples for working with the driver and writing documentation. The last task which I conformed was the study of realization for damage detection which result should be appropriate and realizable method for damage detection in light guides.

Key words

ELCOM, a.s; National Instruments; LabVIEW; servo drive; driver; Modbus; TCP/IP; WorkBench; light guide

Seznam použitých zkratk a symbolů

Skratka	Význam	Bližší popis
ADU	Application Data Unit	základný rámec protokolu Modbus
AKD		registrovaná ochranná známka spoločnosti Kollmorgen TM
CLD	Certified LabVIEW Developer	úroveň certifikácie
CRC	Cyclic Redundancy Check	kotrola cyklickým kódom
DAQ	Data Acquisition	proces zberu dát
HMI	Human-machine interface	užívateľské rozhranie
ISO/OSI	International Organization for Standartization/Open Systems Interconnection	referenčný model siete
IP	Internet Protocol	internetový protokol
LabVIEW	Laboratory Virtual Instruments Engineering Workbench	grafické vývojové prostredie pre návrh systémov
LCD	Liquid-crystal display	displej z tekutých kryštálov
LED	Light-emittion diode	svetelná dioda
MAC	Media Access Control	identifikátor sieťového zariadenia
NI	National Instrument	
PDU	Protocol Data Unit	časť rámca protokolu Modbus
PIN		vstupná jednotka
PLC	Programmable Logic Controller	programovateľný logický automat
POUT		výstupná jednotka
TCP	Transmission Control Protocol	protokol transportnej vrstvy
VI	Virtual Instrument	virtuálny prístroj

Obsah

Úvod.....	- 10 -
1 Firma	- 11 -
1.1 Opis odborného zamerania firmy.....	- 11 -
1.2 Členenie firmy - divízie.....	- 11 -
1.3 Pracovné zaradenie.....	- 11 -
2 Vykonávanie praxe.....	- 13 -
2.1 Zoznam zadáných úloh.....	- 13 -
2.2 Popis a postup riešenia zadáných úloh a využité prostriedky	- 13 -
2.2.1 Vývojové prostredie LabVIEW	- 13 -
2.2.2 Úlohy zamerané na vývoj v LabVIEW.....	- 14 -
2.2.3 Prístrojový ovládač pre servopohon	- 16 -
2.2.4 Štúdia realizovateľnosti	- 22 -
3 Získané znalosti	- 24 -
3.1 Využité teoretické a praktické znalosti získané v priebehu štúdia	- 24 -
3.2 Znalosti a schopnosti chýbajúce v priebehu odbornej praxe	- 24 -
4 Celkové zhodnotenie a záver	- 25 -
Použitá literatúra.....	- 25 -
Zoznam príloh	- 27 -

Úvod

Pre vypracovanie bakalárskej práce som zvolila spôsob absolvovania odbornej praxe vo firme ELCOM, a.s., v divízií Virtuální Instrumentace. Ponúkaná prax bola zameraná na spoluúčasť pri tvorbe projektov, účasť pri vývoji softvérových komponentov, testovanie softvérových komponentov, priebežnú tvorbu dokumentácie a užívateľských návodov. Dôvodom pre výber praxe bola možnosť nadobudnúť znalosti v daných oblastiach a skúsenosti v pracovnom prostredí firmy. Bakalárska práca obsahuje súhrn úloh a ich vypracovanie spolu s výsledkami a závermi, ku ktorým som dospela počas 50 dní praxe. V elektronickej prílohe prikladám kód vytvoreného ovládača, ktorého základné funkcie a popis jeho tvorby je obsiahnutý v kapitole Prístrojový ovládač pre servopohon. Priložený kód obsahuje vlastnú dokumentáciu a ukážkové príklady pre prácu s ovládačom.

1 Firma

1.1 Opis odborného zamerania firmy

Firma ELCOM, a.s. bola založená v roku 1990. Pôvodnou myšlienkou pri jej založení bolo vytvoriť inžinierske a konzultačné stredisko v obore silnoprúdovej elektrotechniky špecializované na oblasť kompenzácie jalového výkonu, elektromagnetickej kompatibility a optimalizáciu spotreby elektrickej energie. V súčasnosti firma ELCOM, a.s. podniká v oblasti silnoprúdovej elektrotechniky, energetiky, virtuálnej instrumentácie a je organizačne rozdelená do piatich divízií a útvarov zaisťujúcich obchodovanie, organizačnú správu a samotné účtovníctvo firmy [1].

1.2 Členenie firmy - divízie

Divízia Realizace a inženýring - venuje sa dodávkam rozvodných a kompenzačných zariadení nízkeho a vysokého napätia s dôrazom na zaistenie elektromagnetickej kompatibility celého výkonového systému. Taktiež sa zaoberá tvorbou rozborov technických stavov napájacích sietí najmä z hľadiska optimálnej prevádzky a zaistenia elektromagnetickej kompatibility.

Divízia Pohony - zameriava sa na pohony v prostredí s nebezpečenstvom výbuchu, v energetike, vodárňach, v ťažobnom priemysle a iných. Je systémovým integrátorom firmy Siemens pre dodávky motorov do prostredia s nebezpečenstvom výbuchu.

Divízia Aplikovaná elektronika - je zameraná na výskum, vývoj a výrobu špeciálnych výkonových elektronických zariadení. Patrí pod ňu výroba zdrojov pre skúšobne a laboratória, špeciálnych napájacích zdrojov, tvorba zdrojov za účelom prispôbiť napájacie technológie pre inú napäťovú sústavu, jednosmerných a pulzných zdrojov. Všeobecne sa dá povedať, že v Divízii Aplikovanej elektroniky je možné vyrobiť ľubovoľný zdroj z akéhokoľvek vstupného napätia a kmitočtu na ľubovoľné napätie, kmitočet a počet fáz na výstupe a to vo výkonoch 0,1 kVA až 1000 kVA.

Divízia Virtuální instrumentace - činnosť firmy sa zameriava na projekty systémovej integrácie najmä v oblastiach automatizovaných meracích a testovacích systémov, systémov monitoringu kvality elektriny, priemyslových testerov a kamerových systémov.

Divízia Výroba - slúži ako výrobný závod pre ostatné divízie a aj ako materiálno-logistická centrála firmy. Výroba je zameraná predovšetkým na rozvádzače nízkeho a vysokého napätia a na kompenzačné rozvádzače. Uskutočňuje taktiež výrobu ochranných tlmiviek do kompenzátorov a úzko spolupracuje s divíziou Aplikovanej elektroniky v oblasti špeciálnych výkonových elektronických zariadení.

1.3 Pracovné zaradenie

Prax som vykonávala v Divízii Virtuální instrumentace, ktorá pôsobí už od roku 1997 vo Vedecko-technologickom parku Ostrava. Divízia je rozdelená na samostatné oddelenia v oblastiach: kamerové systémy, automatizované testery, automatizované skúšobne, meranie kvality energie, konštrukcie špeciálnej elektroniky, strojné konštrukcie, softvérové aplikácie, vývoj nových technológií a marketing.

Moje pracovné zaradenie spadalo pod oddelenie Systémy monitoringu a analýzy kvality elektriny, zameriavajúceho sa na testovanie meracích prístrojov, kalibráciu prístrojov, správu softvéru,

vývoj v LabVIEW a všeobecne prácou s meracími prístrojmi. Práca na projekte ovládača pre servopohon spadala čiastočne pod oddelenie Softvérové aplikácie a prístrojové ovládače. Jednou z oblastí zamerania oddelenia je vývoj prístrojových ovládačov vytváraných pomocou vývojových nástrojov firmy National Instruments, ktorá ponúka ucelenú softvérovú platformu pre vývoj a testovanie aplikácií v oblasti meracej a riadiacej techniky. Štúdia realizovateľnosti bola projektom pre oddelenie Automatizované priemyslové systémy. Cieľom štúdie malo byť nájdenie efektívneho a realizovateľného spôsobu detekcie chýb vo svetlovodoch a jej výsledky by bolo možné uplatniť vo vývoji nástroja pre testovanie svetlovodov.

2 Vykonávanie praxe

Prvou úlohou počas vykonávania praxe bolo zoznámiť sa s vývojovým prostredím LabVIEW a získať potrebné schopnosti pre vývoj v danom prostredí. Najskôr som sa venovala štúdiu dostupnej firemnej literatúry k vývojovému prostrediu a cvičným príkladom, ktoré obsahovala [6-7]. Popri tom som dostávala i zložitejšie príklady, pre overenie nadobudnutých vedomostí. Následne som dostala možnosť pracovať na projekte tvorby prístrojového ovládača pre servopohon. Dodatočne som vypracovala štúdiu realizovateľnosti, ktorej cieľom bolo nájsť vhodnú a realizovateľnú metódu detekcie chýb vo svetlovodoch.

2.1 Zoznam zadaných úloh

Úlohy zamerané na vývoj v LabVIEW.

Vytvorenie TCP/IP komunikácie.

Vlastné riešenie príkladu Car Wash Controller.

Prístrojový ovládač Servo AKD.

Zoznámenie sa s tvorbou ovládača pre servopohon.

Štúdium dokumentácie.

Práca s nástrojom WorkBench.

Tvorba ovládača.

Tvorba ukázkových príkladov pre prácu s ovládačom.

Štúdia realizovateľnosti.

2.2 Popis a postup riešenia zadaných úloh a využité prostriedky

Programovacie prostredie LabVIEW bolo mojim základným pracovným prostredím pre vývoj softvéru a väčšinu nástrojov a prostriedkov som využívala práve v rámci spolupráce s ním. Z prvých náročnejších príkladov, ktoré som vytvorila v LabVIEW uvádzam príklad vytvorenia TCP/IP komunikácie a vlastné riešenie príkladu Car Wash Controller. Ich vypracovanie spolu so zadaním je obsiahnuté v prílohe práce. Po dokončení úloh zameraných na vývoj v LabVIEW som pristúpila k tvorbe prístrojového ovládača pre servopohon AKD. Mojim cieľom bolo splnenie funkčných požiadaviek a tým umožniť predanie ovládača do ďalšej fázy projektu. Pri práci na štúdiu realizovateľnosti som používala dostupné vybavenie oddelenia (svetelné zdroje, držadlá, kamery) a softvérový balíček Pylon pre prácu s kamerami [5].

2.2.1 Vývojové prostredie LabVIEW

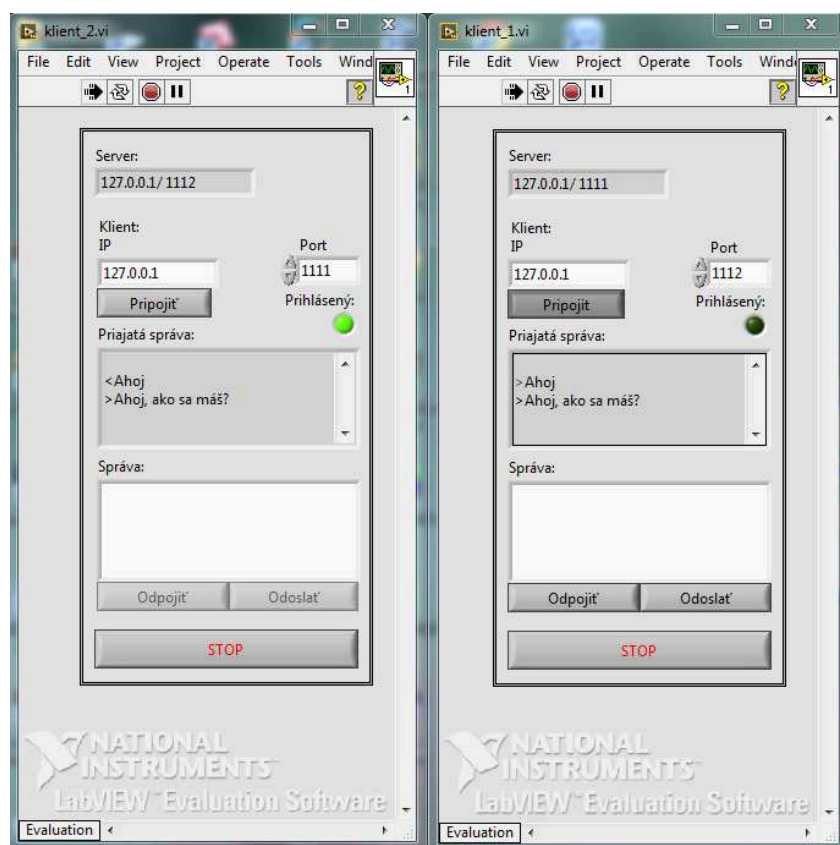
LabVIEW je vývojové prostredie, produkt americkej firmy National Instruments, ktoré využíva plnohodnotný grafický programovací jazyk – tzv. G jazyk. LabVIEW umožňuje vznik aplikácií vhodných na meranie, testovanie, zber dát, riadenie prístrojov, zaznamenanie dát i analýzu meraní. Jedná sa o prostredie vhodné pre prácu so signálom, riadenie a vizualizáciu technologických procesov. Rozdielom medzi LabVIEW a inými textovo orientovanými programovacími nástrojmi, kde inštrukcie určujú vykonávanie programu, je použitie programovania toku dát, kde tok dát určuje vykonávanie kódu. V LabVIEW je budované užívateľské rozhranie, obsahujúce ovládacie prvky a

indikátory – predný panel. Kód, obsahujúci grafickú reprezentáciu funkcií ovládacích prvkov a indikátorov predného panela, predstavuje blokový panel. Je typický svojou schematickou podobou a jedná sa vlastne o algoritmus programu. Program, ktorý je vytvorený pomocou LabVIEW reprezentuje virtuálny prístroj – VI, pomenovanie vyplývajúce z jeho vzhľadu. LabVIEW je súčasťou virtuálnej inštrumentácie, pretože poskytuje ľahko použiteľné prostredie pre vývoj aplikácií a umožňuje ľahké pripojenie dostupných hardvérových i softvérových prostriedkov a prácu s nimi. V oblasti virtuálnej inštrumentácie je cieľom nahradiť meracie prístroje v hardvérovej podobe riešením vytvorenia softvéru, programovateľného nástroja.

2.2.2 Úlohy zamerané na vývoj v LabVIEW

Vytvorenie TCP/IP komunikácie

Výsledná aplikácia je schopná vytvoriť spojenie s využitím TCP komunikačného protokolu medzi 2 klientami. Klient má možnosť zadať IP adresu a port, s ktorým bude komunikovať prostredníctvom textových správ (obrázok 2.1). Vo výslednej aplikácii beží v samostatných cykloch klient a server. V časti klienta je uskutočnené nadviazanie spojenia a zápis správy. Server počúva na definovanom porte prichádzajúce informácie. Cieľom pri tvorbe klienta a servera bolo naučiť sa pracovať s LabVIEW protokolovou knižnicou TCP.



Obrázok 2.1: Predný panel oboch komunikujúcich klientov

TCP/IP

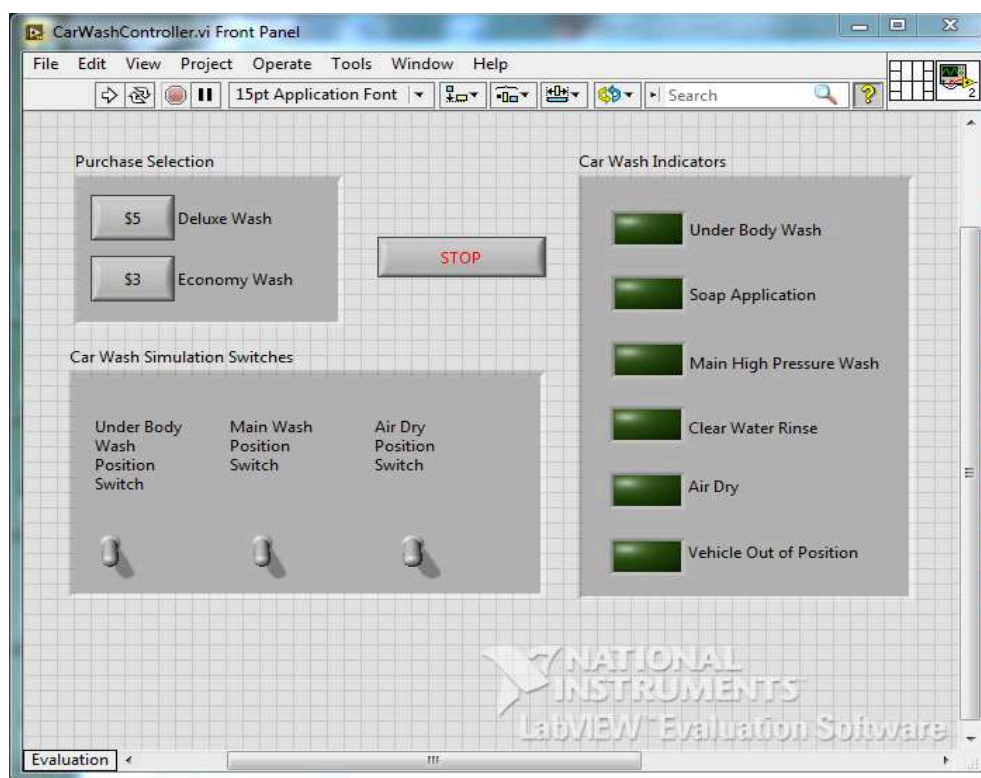
Transmission Control Protocol (TCP) spolu s Internet Protokol (IP) a User Datagram Protocol (UDP) sú základnými prostriedkami pre sieťovú komunikáciu. TCP delí dáta do paketov pre IP vrstvu, odosiela ich a taktiež prevádza kontrolu dát pre uistenie, že dáta prišli do cieľa v jednotnej, kompletnej

podobe. Prostredníctvom TCP/IP je možná komunikácia v jednotlivých alebo prepojených sieťach, v Internete. TCP/IP komunikácia poskytuje jednoduché užívateľské rozhranie, ktoré zakrýva zložitosť vytvorenia a zabezpečenia spoľahlivej sieťovej komunikácie. Dáta pozostávajú z 20 bajtov IP informácií, nasledovaných 20 bajtami TCP informácií, nasledovanými dátami, ktoré boli odoslané.

TCP/IP protokol môže byť použitý vo všetkých platformách LabVIEW. Protokolová funkčná knižnica v LabVIEW poskytuje TCP/IP pomocné funkcie – otvorenie spojenia, čítanie a zápis dát, ukončenie spojenia. Sú tu taktiež VI na prácu s referenciou a IP adresou. U väčšiny I/O komunikácií, procesor je vždy klient, ktorý otvára spojenie k disku vonkajšieho servera, k vonkajšiemu serveru prístroja alebo k DAQ doske servera. S TCP/IP spojeniami, počítač môže fungovať oboma spôsobmi ako klient i ako server alebo démon, počúva na volanie o vzdialené pripojenie a reaguje adekvátnym spôsobom.

Vlastné riešenie príkladu Car Wash Controller

Príklad Car Wash Controller je jeden z testovacích príkladov od NI, pre absolvovanie CLD skúšky. NI ponúka viacero stupňov certifikovania pre LabVIEW programátorov. LCD je stredná úroveň LabVIEW certifikácie. Podľa kritérií zadania vytvorená aplikácia slúži pre riadenie bezkontaktného mytia áut. Bezkontaktné mytie áut pracuje za pomoci vodnej sprchy, bez použitia mycích kef. Aplikácia simuluje prechod, pohyb auta cez myciu linku a vykonávanie jednotlivých cyklov mytia – podvozkové mytie, nanášanie mycích prostriedkov, tlakové mytie, oplachovací proces, sušenie (obrázok 2.2). Každý cyklus mal presne stanovené trvanie a vstup do cyklov bol určený podľa voľby druhu mytia. V mojom prípade bolo cieľom a zámerom pochopiť štruktúru pri vytváraní komplexnej aplikácie a výber efektívnych programovacích prostriedkov. Pracovala som s lokálnym, globálnym typom premenných a funkciami pre časovanie. Snažila som sa správne použiť základné LabVIEW funkcie a bežné programovacie návrhové vzory.



Obrázok 2.2: Predný panel aplikácie Car Wash Controller

2.2.3 Prístrojový ovládač pre servopohon

Mojou úlohou bolo vytvoriť ovládač, ktorý by mal uľahčiť integráciu servopohonu AKD do stávajúceho systému a vytvorený ovládač by mal byť i univerzálne použiteľný v ľubovoľnej aplikácii na báze LabVIEW. Servopohon AKD dovoľuje použitie niekoľkých rozhraní, no ovládač by mal pracovať výhradne s rozhraním Ethernet.

Medzi funkčné požiadavky kladené na ovládač patrilo nasledovné:

Inicializácia zariadenia s možnosťou nastavenia IP adresy a ďalších potrebných parametrov pre naviazanie komunikácie.

Vyčítanie stavu servopohonu.

Vyčítanie aktuálnej uhlovej pozície.

Nastavenie relatívnej a absolútnej pozície v požadovanom rozsahu.

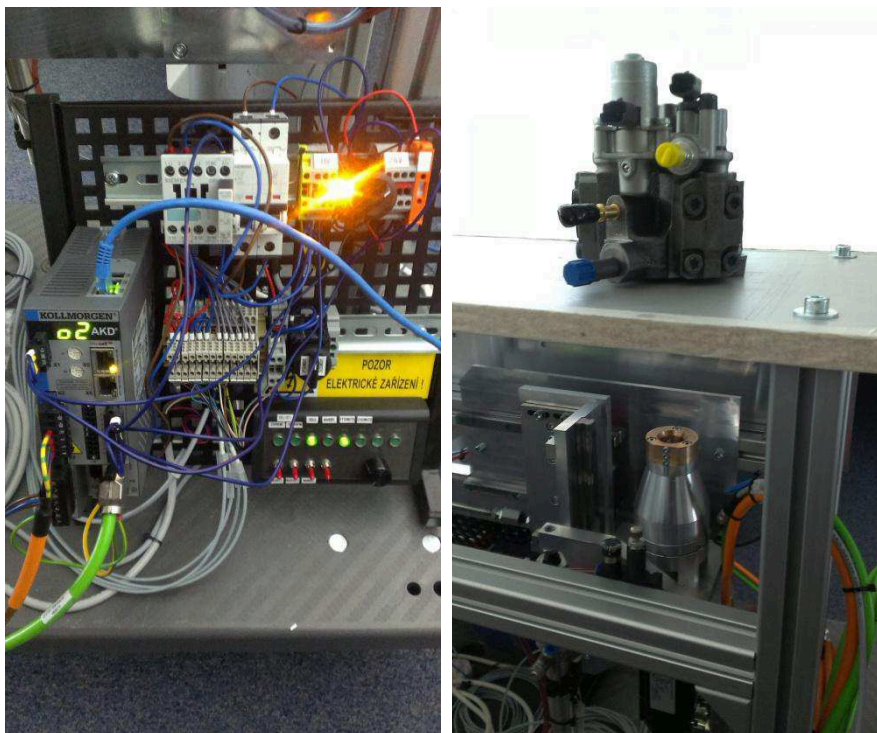
Ukončenie spojenia.

Bezstavovosť.

Schopnosť ovládať ľubovoľný počet podporovaných zariadení.

Práca na projekte

Na začiatku práce som bola zoznámená s tímom kolegov spolupracujúcich na danom projekte a s dokumentáciou určenou pre servopohon AKD. Dokumentácia k servopohonu bola veľmi obsiahla, z jej obsahu som sa zamerala hlavne na časti zamerané na komunikáciu so zariadením [2] a po osvojení si práce s ňou na jednotlivé funkčné príkazy [3]. Pracovala som s funkčným zapojením servomotoru AKD v prototypu modelu zariadenia, pre ktorý bol systém určený (obrázok 2.3).



Obrázok 2.3: Harvérová inštalácia zariadenia

Použitý servomotor je rotačnou akčnou jednotkou, ktorá umožňuje presné riadenie uhlovej pozície. Pozostáva z motora spojeného so senzorom pre pozičnú spätnú väzbu. Kompletný systém obsahuje i servopohon. Pohon používal spätnú väzbu zo senzora pre presné riadenie rotačnej pozície motora.

Základom práce so servomotorom bolo nadviazanie komunikácie, overiť konektivitu a funkčnosť modelu. Pre nastavenie komunikácie so servopohonom bolo určené konkrétne ethernetové rozhranie na jednotke AKD servopohonu a bola potrebná inštalácia softvéru WorkBench.

WorkBench

Nástroj Kollmorgen WorkBench je určený pre priamu komunikáciu s jednotkou servopohonu AKD a pre testovanie zariadení zo série AKD servopohonov. Predstavuje grafické užívateľské rozhranie pre jednoduché nastavenia a konfiguráciu. Nastavenia parametrov pomocou softvéru WorkBench je prevádzané cez TCP/IP (rýchlosť kanála 100 Mbit/s). WorkBench sa snaží nadviazať spojenie so zariadením v sieti a používa discovery protokol pre nájdenie všetkých dostupných zariadení. Ethernet vyžaduje, aby každé zariadenie v sieťovom segmente malo unikátnu IP adresu. IP adresa servopohonu slúži ako unikátny identifikátor zariadenia v sieti a každé AKD má výrobcom danú unikátnu nemennú MAC adresu, ktorá je v Ethernete globálnym unikátnym identifikátorom. WorkBench a zariadenie sa dokážu nájsť na základe IP adresy. V prípade, že viac ako jedno zariadenie je pripojené v sieti, pre potvrdenie komunikácie so správnym zariadením je možné zobrazit' adresu na LED displeji v prednej časti AKD jednotky alebo kontrolou MAC adresy vo WorkBench zozname.

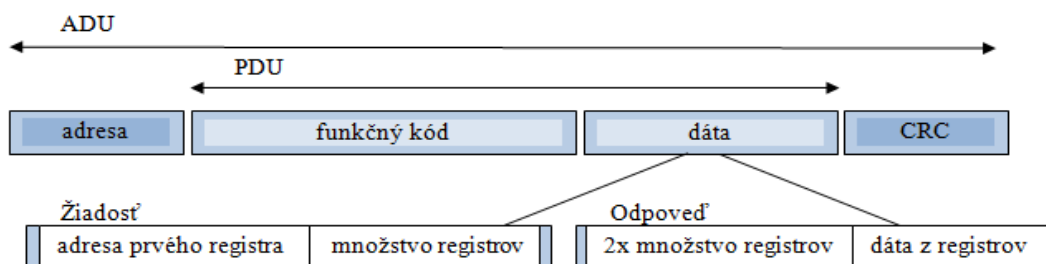
Po nájdení zariadenia v sieti a úspešnom pripojení bolo možné na počítači prostredníctvom nástroja WorkBench konfigurovať a testovať zariadenie. Užitočná bola možnosť programovania v terminálovej časti nástroja, v zmysle priameho využívania príkazov pre konfiguráciu. Po oboznámení sa s funkciami servopohonu, operačnými módmí, parametrami spätnej väzby a ďalšími konfigurovateľnými parametrami som začala pracovať na návrhu aplikácie.

Tvorba aplikácie

Prvým dôležitým krokom bolo zvoliť správny typ komunikácie, ktorý bol podporovaný zariadením a dostatočný k zabezpečeniu požiadaviek na ovládač určený pre model servopohonu. Najvýhodnejšiu voľbu predstavovala komunikácia prostredníctvom komunikačného protokolu Modbus, ako varianta ľahko vytvoriteľná v LabVIEW, len za pomoci TCP/IP komunikačných funkcií a podporovaná modelom zariadenia. Alternatívnou možnosťou bola Ethernet/IP komunikácia.

Modbus

Modbus je komunikačný protokol aplikačnej siedmej vrstvy modelu ISO/OSI [8]. Poskytuje klient/server komunikáciu medzi prepojenými zariadeniami v rôznych sieťach alebo typoch zberníc. Môže byť implementovaný v typoch komunikácie: TCP/IP Ethernet, asynchrónny sériový prenos cez rôzne médiá, Modbus Plus - vysokorýchlostná sieť. Modbus je de facto priemyselný štandard sériovej komunikácie od roku 1979 hlavne pre automatizačné zariadenia. Má jednoduchú štruktúru (obrázok 2.4).



Obrázok č.2.4: Štruktúra Modbus rámca a formát žiadosti a odpovede v dátovej časti pri čítaní pomocou Modbus protokolu

Jedná sa o request/reply protokol a ponúka služby špecifikované vo funkčných kódach, základných prvkoch o veľkosti jedného bajtu, protokolovej dátovej jednotky (PDU), ktorá je nezávislá na nižších komunikačných vrstvách. Modbus využíva prístup k registrom, pre prácu s dátami a presný formát žiadosti a odpovede je určený zvoleným funkčným kódom. Mapovanie Modbus protokolu na konkrétnu zbernicu alebo sieť môže vyžadovať zloženie polí do aplikačnej dátovej jednotky (ADU). Modbus aplikáčna dátová jednotka je vytvorená klientom, ktorý zahajuje Modbus prenos. Pre Modbus aplikácie je rezervovaný systémový port 502 v TCP/IP komunikácii. Typy zariadení ako PLC, HMI, Control Panel, Driver, Motion control, I/O Device môžu použiť Modbus protokol pre iniciáciu vzdialeného riadenia. Rovnaké možnosti umožňuje komunikácia cez sériovú linku ako i cez Ethernet TCP/IP sieť.

Modbus TCP komunikácia so zariadením bola umožnená sieťovým rozhraním servopohonu používaným i pre komunikáciu s nástrojom WorkBench. Modbus komunikácia so servopohonom prebiehala na porte 502 a paralelná komunikácia s nástrojom WorkBench na rozdielnom porte. Pri budovaní Modbus ADU bolo nutné zohľadniť, že servopohonom nie je podporovaný CRC kód. Dve funkcie sú podporované pre Modbus komunikáciu: Funkcia Read Holding Registers s funkčným kódom 0x03 a Write Multiple Registers s funkčným kódom 0x10. Funkcie dovoľujú len 32-bitový a 64-bitový prístup. Množstvo registrov je limitovaných na 2 pre 32-bitové premenné a 4 registre pre 64-bitové premenné. Je možné taktiež čítanie/zápis do jednoduchého registra pre určité parametre. Iný počet registrov vracia chybové hlásenie. AKD servopohon dokáže spracovať s využitím Modbusu väčšinu príkazov a je ním možné plne ovládať zariadenia. Výnimku z použiteľných príkazov predstavujú tie, ktorých výstupom je reťazec alebo skupina reťazcov.

Po testovaní Modbus príkazov v nástroji WorkBench a sledovaní reakcií servopohonu som získala prehľad o parametroch, ktoré boli nevyhnutné pri tvorbe ovládača. Jednalo sa o aktuálnu pozíciu – hodnoty spätnej väzby, operačný mód, parametre operačného módu a nastavenie pozičných jednotiek. Spočiatku nastal problém s hodnotami uloženými v registroch, kvôli tomu, že dáta sú ukladané v pamäti prístroja endianitou big-endian a bolo potrebné prispôbiť pôvodné Modbus funkcie pre správne čítanie z registrov a zápis do registrov pri všetkých typoch premenných.

Vhodný operačný mód pracujúci s mechanickým pohybom motora na špecifikovanú pozíciu, predstavoval v nástroji WorkBench mód Homing. Operačný mód Homing pracuje s parametrom pozície ako s referenčným bodom predstavujúcim známu východiskovú pozíciu a ponúkal viacero metód pre dosiahnutie tejto pozície. Mód umožňoval nastaviť aktívne možnosti v závislosti na zvolenej metóde ako akceleráciu, deceleráciu, smer, vzdialenosť, pozíciu, rýchlosť, kladný a záporný limitový spínač. Umožňuje i sledovať stav pohybu - východisková pozícia bola nájdená, pohyb bol

dokončený alebo je stále aktívny, zlyhanie pohybu. Prítomný je aj indikátor s hodnotou pozičnej spätnej väzby a riadenie na zahájenie pohybu a zastavenie.

Pre účely servopoponu bolo najvhodnejšie zvoliť základnú operačnú Homing metódu Use current position. V metóde aktuálnej pozície dve parametrové hodnoty slúžia pre riadenie: vzdialenosť a pozícia. Hodnota parametru vzdialenosť iná ako nula spôsobí pohyb motora o vzdialenosť zadanú v nastavených jednotkách po zahájení pohybu. Je možné fixne nastaviť vzdialenosť od východiskovej pozície motora. Parameter pozície slúži pre nastavenie hodnoty východiskovej pozície rozdielnej od nuly. Aktuálna spätná väzba sa po zahájení pohybu bude meniť, pokiaľ motor nedosiahne východiskovú pozíciu.

Nájdenie spätnej väzby, ktorá by zodpovedala aktuálnej pozícii bolo spojené s viacerými problémami. Prvým bol prevodový stupeň medzi reálnym uhlom natočenia a spätnou väzbou. Aktuálna hodnota spätnej väzby sa rovnala desaťnásobku hodnoty skutočnej mechanickej pozície - uhla natočenia. Nenašla som správny spôsob pre manipuláciu s prevodovým stupňom, no ako jedno z riešení bolo možné vyčítať spätnú väzbu obsahujúcu absolútnu pozíciu natočenia. Táto hodnota zodpovedala celkovému počtu otočení bez spracovania a prepočtov. Menším problémom boli jednotky spätnej väzby, pretože jedno otočenie predstavovalo hodnotu 4 294 967 296 a manipulácia s jednotkami v danom rozsahu bola málo prehľadná. Alternatívne bolo možné nastaviť užívateľsky definované jednotky pre daný parameter. To sa ukázalo ako vhodné riešenie spolu s nastavením parametrov vstupnej jednotky na 360 a výstupnej jednotky na 1. Priamo v pohone sa uskutočnil prepočet hodnôt. Výsledná hodnota sa rovnala počtu stupňov otočenia, ktoré motor uskutočnil od posledného spustenia zariadenia. Vo vytvorenom ovládači sa hodnota ďalej spracovávala aby zodpovedala uhlovej pozícii natočenia v rozsahu 0-360 stupňov. Potrebný bol prepočet otočení motora a ošetrovanie zápornej hodnoty v parametre. Ak by hodnota absolútnej spätnej väzby prekročila nulovú hodnotu došlo by k vyčítaniu záporného čísla a ovládač by zamedzil možnosť nastavenia pozície.

Nastavenie jednotiek bolo možné pre parametre pohybu: pozícia, rýchlosť a akcelerácia. Spolu s jednotkami súvisia i parametre škálovania, prepočtu a detailu mechanického systému – rotačný typ motora pracoval s rotačnými jednotkami. Motor používa defaultne nastavené jednotky Counts. Sú najmenšou jednotkou pozície, ktoré mohol pohon používať. Jednotka zodpovedá 4 294 967 296 Counts/rev - na otáčku motora. Ďalšie možnosti pre výber jednotky boli radiány ($2\pi/\text{rev}$), stupne ($360/\text{rev}$) a užívateľom definovaná jednotka (PIN/POUT). Použitím užívateľom definovanej jednotky bolo možné zohľadniť mechanizmus zariadenia, zadaním vstupnej a výstupnej hodnoty jednotky prepočtu na otáčku. Voľba jednotiek neovplyvňovala prevod medzi jednotkami alebo presnosť pohonového systému. Pre pozíciu som zvolila jednotku stupne, šlo o rotačný pohyb, v obmedzení 0-360 stupňov. Jednotka bola platná pre vzdialenosť, východiskovú pozíciu a aktuálnu spätnú väzbu. Absolútna spätná väzba mala definovaný vlastný prepočet. Rýchlosť a akcelerácia pracovali s defaultne nastavenými rotačnými jednotkami. Dôležitá bola i voľba škálovania jednotiek nastavená na škálovanie podľa interných jednotiek pohonu, aby sa prenášané hodnoty neupravovali v komunikácii cez Modbus.

Monitorovanie chybových hlásení vznikajúcich počas komunikácie alebo počas pohybu motora je dôležité pre monitorovanie stavu motora a pri tvorbe reakcie na vzniknutú situáciu. Obmedzením pri tvorbe ovládača bolo, že Modbus neumožňoval prenos reťazcov pre zobrazenie správy o chybe. Každá chyba je asociovaná s číslom a vďaka nemu sa dala chyba dohľadať v príslušnej dokumentácii. Vytvorené funkcie v ovládači vracajú číselné kódy chýb.

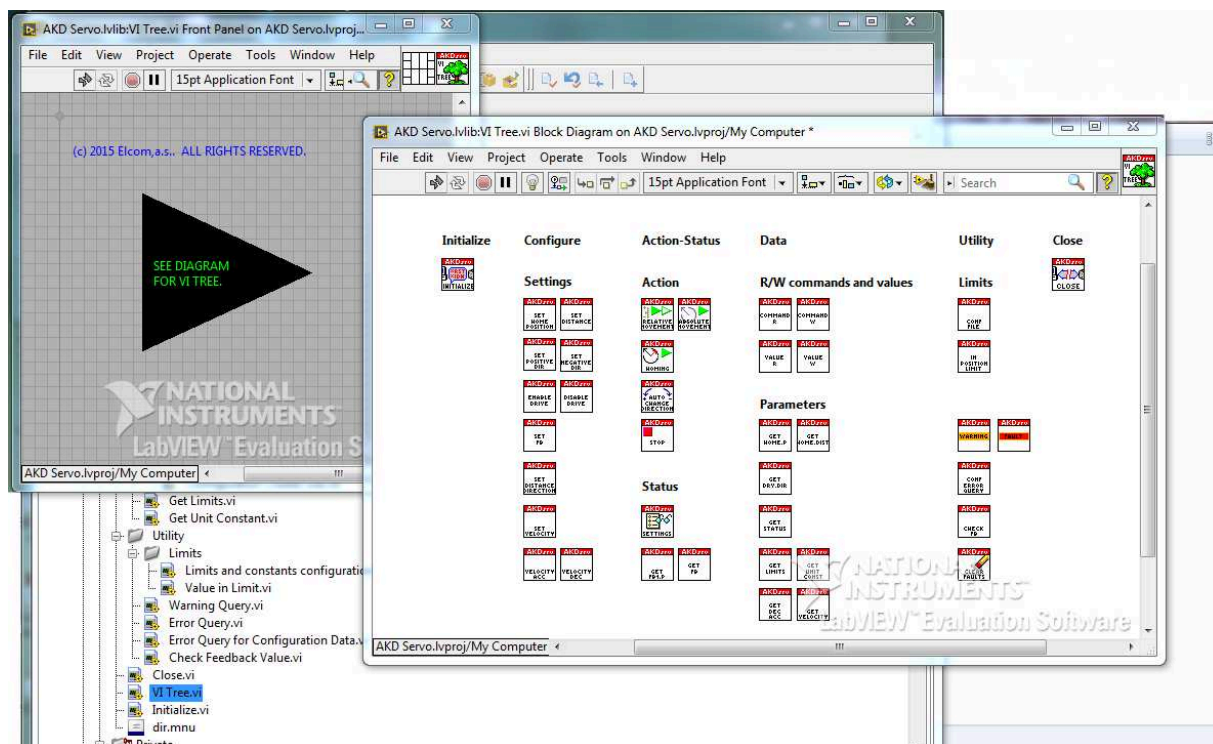
Dodatočne bola pridaná možnosť nastavenia rýchlosti, akcelerácie a decelerácie pohonu. Súviselo to s mechanizmom nájdenia zadanej pozície a hladkého nábehu na dané miesto. Snažila som sa tým zabrániť nadmernému poškodzovaniu časti hlavice, ktorá zapadala do určenej drážky.

Existovala možnosť nastaviť limity v nástroji WorkBench a uložiť ich v pamäti prístroja, no problém predstavovala opäť spätná väzba motora. Limity boli určené pre hodnotu z parametra aktuálnej spätnej väzby a tá zodpovedala desaťnásobku skutočnej hodnoty mechanickej pozície. Preto som limity spracovala priamo v ovládači, obmedzením nastavenia uhlovej pozície v rozmedzí 0-360 stupňov.

Príkaz pre samotný rotačný pohyb bol dostupný vo viacerých operačných módoch a obmedzený parametrami daného módu. V Homing móde a s použitím metódy aktuálnej pozície bol pohyb obmedzený len nastavením aktuálnej vzdialenosti a zahájením pohybu. Zastavenie pohybu bolo možné kedykoľvek počas zahájenia pohybu v móde alebo použitím priameho príkazu pre okamžité zastavenie.

Zariadenie bolo pomocou patričných príkazov možné uviesť do stavu, keď bol umožnený manuálny pohyb otočnou časťou a status indikoval neaktívnosť polohového systému. V aktívnom stave bol manuálny pohyb pohonom znemožný a rotačná poloha pohonu bola riadená len softvérovo.

Informácie o hodnotám pozičných parametrov boli prístupné priamo čítaním z Modbus registrov. Čítať bolo možné informácie o aktuálnej pozícii, smere otočenia, vzdialenosti a hodnote východiskovej pozícii. Všetky vytvorené funkcie, ktoré ovládač poskytuje pre riadenie servopohonu sú organizované v stromovej štruktúre vytvoreného projektu AKD Servo (obrázok 2.5).



Obrázok 2.5: Funkcie pre prácu s ovládačom AKD Servo

Výsledný projekt

Vytvorený ovládač je projektom typu Plug and Play [4]. Ovládač využíva Modbus TCP/IP komunikačný protokol a je možné zadať adresu zariadenia, s ktorým sa snažíme nadviazať spojenie.

Inicializácia zariadenia slúži pre nadviazanie spojenia so zariadením, nájdenie pozície pohonu a voliteľne uskutočniť nastavenie konfiguračných dát alebo nájdenie východiskovej pozície. Konfiguračné dáta obsahujú užívateľom definované limity pre nastavenie rotačnej polohy a jednotkovú konštantu. Jednotková konštantu slúžila na prepočet skutočnej mechanickej polohy pohonu zo spätnej väzby, defaultná hodnota je rovná 10.

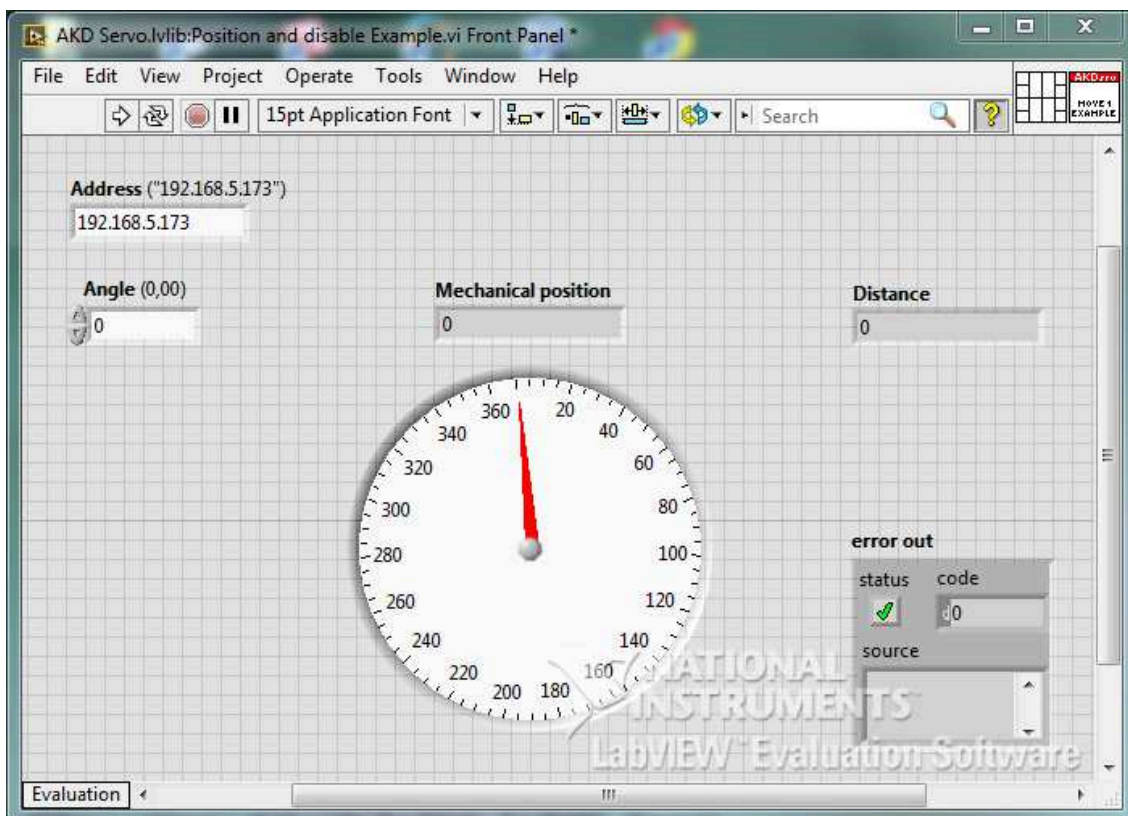
Po úspešnej inicializácii je možná manipulácia alebo vyčítanie hodnôt zo servopohonu. Manipuláciu predstavujú dostupné funkcie motora. Relatívne nastavenie pozície, absolútne nastavenie pozície, nastavenie pozície s uvoľnením pohonu. Je možná i rôzna kombinácia daných funkcií a funkcia okamžitého zastavenia pohybu.

Ovládač umožňuje nie len vyčítanie hodnôt z pohonu, ale taktiež ich nastavenie. Jedná sa o hodnoty východisková pozícia, vzdialenosť uhla otočenia, smer otočenia, rýchlosť rotačného pohybu, akceleráciu a deceleráciu. Rovnako je možné ovládať i aktívny stav pohonu, v ktorom nie je možná manuálna manipulácia s pohonom.

Je možné detekovať prítomnosť chybových hlásení a varovaní priamo zo zariadenia a chyby vyskytujúce sa kvôli nesprávnej manipulácii s ovládačom alebo v dôsledku poruchy v komunikácii.

Limity obmedzujúce nastavenie rotačnej pozície sú vymedzené v rozsahu 0-360 stupňov. Ovládač sa stará i o bezpečné ukončenie nadviazanej komunikácie.

V ďalšej časti vývoja som sa zamerala na ukážkové príklady práce s ovládačom. Keďže som nemala dostatočné skúsenosti s písaním tak rozsiahleho kódu v podobnej štruktúre, snažila som sa vždy po pridaní novej funkcie vytvoriť testovacie VI a overiť si výsledok funkcie. Viaceré z týchto VI som ponechala v projekte ovládača ako ukážkové príklady. Hlavné ukážkové VI dokáže nastaviť absolútnu uhlovú pozíciu servopohonu a prejsť do neaktívneho stavu, ktorý umožní po ďalšom pohybe jemnú manipuláciu s pohyblivou časťou pre presné zapadnutie do drážky. Aktuálna mechanická pozícia je snímaná počas celej doby otočenia. V príklade je ako vstup uvedená IP adresa zariadenia, s ktorým bude nadviazaná komunikácia. Ako výstup je okrem zobrazenia pozície i vzdialenosť otočenia od východiskovej pozície a výstup chybového hlásenia (obrázok 2.6).



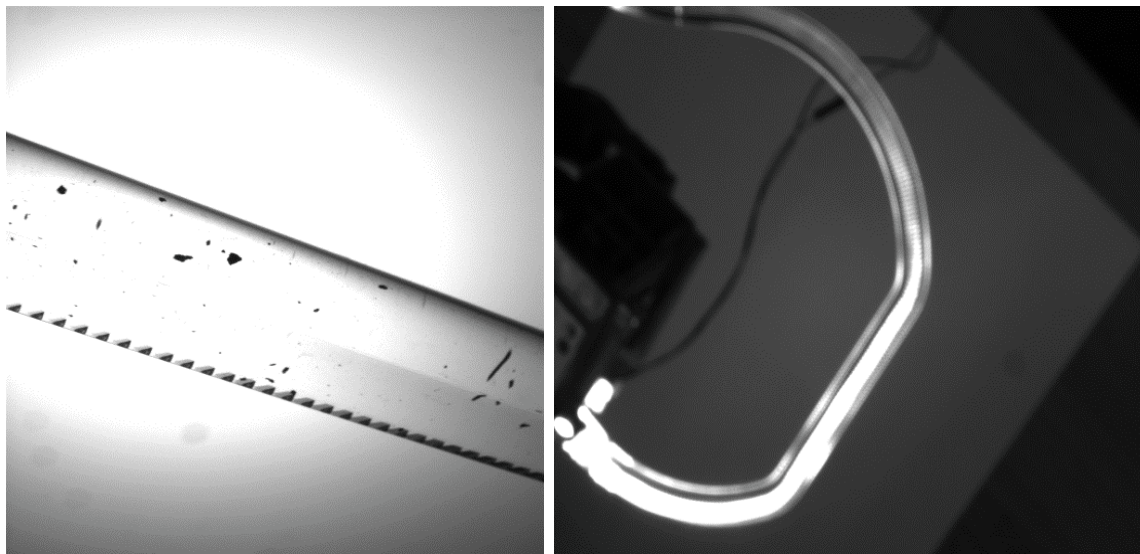
Obrázok 2.6: Predný panel ukážkového príkladu pre manipuláciu so servopohonom

Finálnou časťou pri tvorbe ovládača bola tvorba jeho dokumentácie. Tá zahŕňovala okrem spracovania dokumentácie k jednotlivým VI i dokumentáciu k prístrojovému ovládaču AKD Servo so špecifikáciou a funkciami ovládača.

2.2.4 Štúdia realizovateľnosti

Zámerom štúdie realizovateľnosti bolo nájsť správnu metódu pre detekciu chýb vo svetlovodoch. Pracovala som s dvoma typmi kamier a softvérom pre ich konfiguráciu a riadenie [5]. Dostupné boli taktiež rôzne druhy svetelných zdrojov. Pri testovaní som ako zdroj svetla používala klasické a infračervené LED diody. Svetlovody určené pre testovanie boli rozdelené do skupín, podľa druhu chyby svetlovodu. Testovala som nasledujúce skupiny chybovosti: čierne častice vo svetlovode, bubliny vo svetlovode, zvrásnenie na povrchu svetlovodu a chyby v rebrovej štruktúre svetlovodu. Použité metódy pre umiestnenie kamery vo vzťahu k svetlu a prostredia boli nasledovné: svetlo prenikalo spoza bieleho pozadia, svetlo bolo privádzané priamo do vnútra svetlovodu a kombinácia oboch metód. Jednou z navrhovaných metód bolo podsvietenie svetlovodu pomocou LCD monitora, ktorá však neposkytovala žiadaný výsledok. Monitorom vyžiarené svetlo prenikajúce cez svetlovod bolo síce rozptýlené rovnomerne, čo bolo dôležité kvôli nepravidelnej štruktúre svetlovodu, no jeho intenzita nebola dostatočná. Pre zachytenie chyby boli možné nasledovné typy snímania: priamy pohľad na svetlovod, svetlovod pod zorným uhlom, detailný bočný pohľad. Pôvodne boli metódy snímania priameho pohľadu a zorného uhla navrhnuté z dôvodu, že v reálnom použití svetlovodu sú chyby najviac viditeľné. No pre softvérové spracovanie bola metóda detailného bočného pohľadu najvhodnejšia. Pre skupiny svetlovodov s poruchami čiernych častíc, bublín alebo svetlovodov s povrchovým zvrásnením sa ukázal byť najvhodnejším typom snímania detailný bočný pohľad a

metóda privádzaného svetla spoza bieleho pozadia. Pre chyby v rebrovej štruktúre svetlovodu bolo riešením privádzanie svetla do vnútra svetlovodu a priamy pohľad na svetlovod. (obrázok 2.7).



Obrázok 2.7: Detekovanie čiernych škvŕn (vľavo), detekcia poruchy v rebrovej štruktúre svetlovodu (vpravo)

Ako riešenie pre detekciu chýb vo svetlovode, sa podľa výsledkov štúdie ponúkalo zatiaľ len teoretické riešenie, zostrojiť mechanizmus robota, ktorý bude vykonávať trajektóriu pohybu v tvare svetlovodu s pevne upevnenou snímajúcou kamerou. V testovacom zariadení by bol prítomný zdroj difúzneho svetla, ktorý by bol umiestnený v pozícii proti stojacej snímajúcej kamere.

3 Získané znalosti

3.1 Využité teoretické a praktické znalosti získané v priebehu štúdia

Absolvovaná bakalárska prax sa v konečnom dôsledku zamerala na viaceré oblasti. Znalosti, ktoré som uplatnila pri vytváraní aplikácií v LabVIEW a práci na ovládači som čerpala z predmetov zameraných na programovanie ako Algoritmy či Základy programovania a pri práci s komunikačnými protokolmi som využila poznatky z predmetu Počítačové siete. Užitočné poznatky som získala i z predmetov Virtuálna inštrumentácia a Meranie v informačných a komunikačných technológiách, ktorých obsah sa zameriaval na programovanie v LabVIEW a ktoré obohatili moje vedomosti o mnohé užitočné informácie z daných oblastí.

3.2 Znalosti a schopnosti chýbajúce v priebehu odbornej praxe

Obdobie dochádzania do firmy kvôli bakalárskej praxi zahŕňalo interval dvoch semestrov a mnoho poznatkov, hlavne zo začiatku praxe som získala samoštúdiom. Chýbajúce znalosti sa týkali najmä programovania v LabVIEW. Snažila som sa však využiť ponúkané študijné možnosti a zamerať svoj študijný plán na oblasti, ktoré by boli užitočné pri práci vo firme ELCOM,a.s.. Chýbali mi všeobecné znalosti z elektrických meraní a virtuálnej inštrumentácie.

4 Celkové zhodnotenie a záver

Práca vo firme bola pre mňa veľkým prínosom. Získala som prax a skúsenosti z reálneho pracovného prostredia a poznatky o fungovaní firmy. Naučila som sa aká dôležitá je komunikácia so spolupracovníkmi a kolegami. Počas praxe som objavila pre mňa dovtedy neznáme vývojové prostredie LabVIEW, zoznámila sa so základmi práce s meracími prístrojmi i získala prehľad v oblasti pôsobenia firmy ELCOM, a.s.. I keď som sa štúdiu LabVIEW venovala pri tvorbe mnohých príkladov, najviac poznatkov som nadobudla pri vývoji ovládača. Jednalo sa nielen o písanie kódu ale i riešenie technických problémov, neustále štúdium dokumentácie a hľadanie nie len použiteľných riešení. Ako výsledok mojej práce sa tým stal funkčný produkt – ovládač AKD Servo. Vyskúšala som si i prácu, ktorá nebola zameraná čisto na programovanie a to prácu s kamerovým systémom i keď len v jednoduchej štúdií. Verím, že v budúcnosti budem schopná nadviazať na získané poznatky a uplatniť ich pri ďalšej práci.

Použitá literatura

- [1] ELCOM. O společnosti - ELCOM, a.s. [online]. © 2015 [cit. 2015-04-19]. Dostupné z: <http://www.elcom.cz/cz/spolecnost/o-spolecnosti>
- Virtuální instrumentace I. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2014. ISBN 978-80-248-3395-8.
- [2] KOLLMORGEN. HMI Modbus Communication with AKD [online]. 2012 [cit. 2015-04-19]. Part Number: 903-200010-00. Dostupné z: http://www.kollmorgen.com/zuza/products/drives/servo/akd/_manuals/kollmorgen-akd-hmi-modbus-communications-manual-en-rev-a-fw-1-13/
- [3] KOLLMORGEN. AKD User Guide [online]. 2014 [cit. 2015-04-19]. Part Number: 903-200006-00. Dostupné z: http://www.kollmorgen.com/en-us/products/drives/servo/akd/_manuals/kollmorgen-akd-user-guide-en-rev-n/
- [4] Developing LabVIEW Plug and Play Instrument Drivers - National Instruments. NATIONAL INSTRUMENTS. Plug and Play Instrument Drivers [online]. © 2015 [cit. 2015-04-19]. Dostupné z: <http://www.ni.com/tutorial/3271/en/>
- [5] Pylon. Kamera-Software | Basler [online]. © 2015 [cit. 2015-04-19]. Dostupné z: <http://www.baslerweb.com/de/produkte/software>
- [6] *Virtuální instrumentace I: učební text*. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2014. ISBN 978-80-248-3395-8.
- [7] NATIONAL INSTRUMENTS. LabVIEW: User Manual. 2003. Part Number: 320999E-01.
- [8] Modbus. Modbus protocol [online]. © 2005-2015 [cit. 2015-04-28]. Dostupné z: <http://www.modbus.org/specs.php>

Zoznam príloh

Súčasťou BP je CD.

Štruktúra adresára priloženého CD:

Příloha A: *Prístrojový ovládač AKD Servo.*

Příloha B: *Potrebná inštalácia.*

Příloha C: *Úlohy zamerané na vývoj v LabVIEW.*